<u>**AMENDMENTS TO THE CLAIMS**</u>

This listing of claims will replace all prior versions of claims in the application:

Listing of Claims:

1.   (Currently Amended) A ~~computer implemented~~ system <u>executing one or more processors</u> that facilitates AML <u>(ACPI Machine Language)</u> access to an SMBus <u>(System Management Bus)</u>, comprising the following ~~computer executable~~ components:

an AML event handler component; and,

a driver component that identifies an SMBus event and dispatches the SMBus event to the AML event handler, where the AML event handler employs at least one of a three parameter buffer access read method to read data from an operation region associated with the SMBus [[and]] <u>or</u> a three parameter buffer access write method to write data to an operation region associated with the SMBus.

2.   (Original) The system of claim 1, where the driver receives a status and a data associated with the SMBus event from the SMBus.

3.   (Original) The system of claim 1, where the driver employs a _Qxx control method to dispatch the SMBus event to the AML event handler.

4.   (Currently Amended) The system of claim [[1]] <u>3</u>, where at least one AML event handler entry point is accessed by the _Qxx control method.

5.   (Previously Presented) The system of claim 4, where a first parameter of the three parameter buffer access read method provides an initial data to a computer component providing access to the operation region associated with the SMBus.

6.   (Original) The system of claim 5, where a second parameter of the three parameter buffer access read method is a reference to the operation region associated with the SMBus from which the data will be read.

7.   (Original) The system of claim 6, where a third parameter of the three parameter buffer access read method holds data read from the operation region identified by the second parameter.

8.   (Original) The system of claim 6, where a third parameter of the three parameter buffer access read method is a reference to a location to store the data read from the operation region identified by the second parameter.

9.   (Cancelled)

10.   (Previously Presented) The system of claim 1, where a first parameter of the three parameter buffer access write method is the data to be written to the operation region associated with the SMBus.

11.   (Previously Presented) The system of claim 1, where a first parameter of the three parameter buffer access write method is a reference to the data to be written to the operation region associated with the SMBus.

12.   (Previously Presented) The system of claim 1, where a second parameter of the three parameter buffer access write method is a reference to the operation region associated with the SMBus to which the data will be written.

13.   (Previously Presented) The system of claim 1, where a third parameter of the three parameter buffer access write method is a status code returned by a computer component providing access to the operation region associated with the SMBus.

14.   (Currently Amended) A ~~computer~~ system <u>executing</u> ~~holding computer executable~~ components <u>one or more processors</u> that facilitate access to an SMBus <u>(System Management Bus)</u>, comprising:

    a computer executable identifier that identifies an SMBus event notification at a driver; and

    a computer executable dispatcher in the driver that directly dispatches the SMBus event notification to a computer executable AML <u>(ACPI Machine Language)</u> event handler<u>, where the AML event handler employs at least one of a three parameter buffer access read method to read data from an operation region associated with the SMBus or a three parameter buffer access write method to write data to an operation region associated with the SMBus</u>.

15.   (Currently Amended) A computer implemented method for SMBus <u>(System Management Bus)</u> event handling, the method comprising the following computer executable acts:

    receiving an SMBus event notification at a driver;

    identifying the SMBus event notification;

    dispatching the SMBus event notification to an AML <u>(ACPI Machine Language)</u> event handler<u>, where the AML event handler employs at least one of a three parameter buffer access read method to read data from an operation region associated with the SMBus or a three parameter buffer access write method to write data to an operation region associated with the SMBus</u>; and

    handling the SMBus event notification in AML code.

16.   (Original) The method of claim 15, where the SMBus event notification is identified by examining at least one of a data and a status associated with the SMBus event notification.

17.   (Original) The method of claim 15, where dispatching the SMBus event notification comprises indexing to a _Qxx control method via a registered AML event handler.

18.   (Original) The method of claim 15, where handling the SMBus event notification in AML code comprises reading an operation region associated with the SMBus that generated the SMBus notification.

19.   (Original) The method of claim 18, where the operation region is accessed by a three parameter read, where a first parameter holds an initial data, a second parameter holds a reference to the operation region to be accessed and a third parameter holds data read from the operation region.

20.   (Original) The method of claim 18, where the operation region is accessed by a three parameter read, where a first parameter holds an initial data, a second parameter holds a reference to the operation region to be accessed and a third parameter holds a reference to data read from the operation region.

21.   (Original) The method of claim 15, where handling the SMBus event notification in AML code comprises writing an operation region associated with the SMBus that generated the SMBus notification.

22.   (Original) The method of claim 21, where the operation region is written by a three parameter write, where a first parameter holds a data to be written to the operation region, a second parameter holds a reference to the operation region and a third parameter holds a returned status call.

23.   (Original) The method of claim 21, where the operation region is written by a three parameter write, where a first parameter holds a reference to a data to be written to the operation region, a second parameter holds a reference to the operation region and a third parameter holds a returned status call.

24.   (Previously Presented) A computer system storing computer instructions operable to perform the method of claim 15.

25.   (Currently Amended) A computer executable system for SMBus (System Management Bus) event handling, comprising:

computer implemented means for receiving an SMBus notification via a _Qxx control method;

computer implemented means for locating an AML (ACPI Machine Language) code event handler associated with the SMBus notification; and

computer implemented means for the _Qxx control method to dispatch the SMB notification to the AML code event handler associated with the SMBus notification, where the AML code event handler employs at least one of a three parameter buffer access read method to read data from an operation region associated with the SMBus or a three parameter buffer access write method to write data to an operation region associated with the SMBus.

26.   (Original) The system of claim 25, comprising means for the AML code event handler to access a data object employed to communicate with an SMBus.

27.   (Currently Amended) A data structure stored on a computer readable storage medium employed by computer implemented processes executing on a computer system that facilitates dispatching an SMBus (System Management Bus) event to an AML (ACPI Machine Language) code event handler , the data structure comprising:

at least one indexed AML code entry point; and

at least one AML event handler entry point associated with the at least one indexed AML code entry point.